IN THE CLAIMS:

1. (Currently Amended) In a ~~system comprising a~~ computer subsystem, ~~which includes an electronics assembly engineering system,~~ in which user-defined data structures accessible to editor software have referential integrity, and in which user modifications to the data structures during editing are made directly to the data structures rather than indirectly by way of a temporary file, a method for permitting naming and manipulation of the data structures, the method comprising the steps of:

providing close, discard and rename functions for the data structures, if a newly-created data structure is being edited;

providing close and copy functions for the data structures if an existing data structure is being edited; ~~and~~

ensuring that ~~excluding~~ a save-as function for the data structures is entirely excluded so as to be incapable of being executed, and

employing the computer subsystem in an electronics assembly engineering system.

2. (Original) The method according to claim 1, wherein the data structures comprise objects.

3. (Original) The method according to claim 1, wherein the data structures comprise mark-up language documents.

4. (Original) The method according to claim 3, wherein the data structures comprise XML documents.

5. (Currently Amended) In a system, a computer subsystem in which user-defined data structures accessible to editor software have referential integrity, and in which user modifications to the data structures during editing are made directly to the data structures rather than indirectly by way of a temporary file, the subsystem comprising:

~~an electronics assembly engineering system;~~

a computer-readable media having stored on it instructions for performing naming and manipulation functions for the data structures, the functions comprising (i) close, discard and rename functions for the data structures, if a newly-created data structure is being edited; and (ii) close and copy functions for the data structures if an existing data structure is being edited; and the functions entirely excluding ~~not comprising~~ a save-as function for the data structures so as to be incapable of being executed.

6. (Original) The subsystem according to claim 5, wherein the data structures comprise objects.

7. (Original) The subsystem according to claim 5, wherein the data structures comprise mark-up language documents.

8. (Original) The subsystem according to claim 7, wherein the data structures comprise XML documents.

9. (Original) The subsystem according to claim 5, wherein the computer-readable media is removable from the subsystem.

10. (Currently Amended) A computer-readable media of ~~comprising an electronics assembly engineering system comprising~~ a computer subsystem in which user-defined data structures accessible to editor software have referential integrity, and in which user modifications to the data structures during editing are made directly to the data structures rather than indirectly by way of a temporary file, the media having stored on it instructions for performing a method for permitting naming and manipulation of the data structures, the method comprising the steps of:

providing close, discard and rename functions for the data structures, if a newly-created data structure is being edited;

providing close and copy functions for the data structures if an existing data structure is being edited; and

ensuring that ~~excluding~~ a save-as function for the data structures is entirely excluded so as to be incapable of being executed.

11. (Currently Amended) A computer-readable media of ~~comprising an electronics assembly engineering system comprising~~ a computer subsystem in which user-defined data structures accessible to editor software have referential integrity, and in which user modifications to the data structures during editing are made directly to the data structures rather than indirectly by way of a temporary file, the media having

stored instructions for performing naming and manipulation functions for the data structures, the functions comprising (i) close, discard and rename functions for the data structures, if a newly-created data structure is being edited; and (ii) close and copy functions for the data structures if an existing data structure is being edited; and the functions entirely excluding ~~not comprising~~ a save-as function for the data structures so as to be incapable of being executed.

12. (Currently Amended) In a computer system in which user-defined data structures accessible to editor software have referential integrity, and in which user modifications to the data structures during editing are made directly to them rather than indirectly by way of a temporary file, a method for permitting naming and manipulation of the data structures, the method comprising the steps of:

providing close, discard and rename functions for the data structures, if a newly-created data structure is being edited;

providing close and copy functions for the data structures if an existing data structure is being edited; and

ensuring that ~~excluding~~ a save-as function for the data structures is entirely excluded so as to be incapable of being executed.

13. (Original) The method according to claim 12, wherein the data structures comprise objects.

14. (Original) The method according to claim 12, wherein the data structures comprise mark-up language documents.

15. (Original) The method according to claim 14, wherein the data structures comprise XML documents.

16. (Currently Amended) A method for enabling data structure naming and manipulation functions in a computer system coupled to a display and employing transacted service, wherein the data structures have referential integrity and temporary copies of data structures are not created during editing of the data structures, the method comprising the steps of:

presenting on the display a representation of a plurality of data structures; and

providing a plurality of functions for either or both of naming and manipulation of data structures, the plurality of functions entirely excluding a save-as function so as to be incapable of being executed.

17. (Original) The method according to claim 16, wherein the plurality of manipulation functions comprises providing close, discard and rename functions if a newly-created data structure is being edited.

18. (Original) The method according to claim 16, wherein the plurality of manipulation functions comprises providing close and copy functions if an existing data structure is being edited.

19. (Original) The method according to claim 16, wherein the step of presenting on the display a representation of a plurality of data structures comprises presenting a graphical representation of a plurality of data structures.

20. (Currently Amended) A method for enabling naming and manipulation functions for data structures in a computer subsystem ~~comprising an electronics assembly system engineering system~~, the computer subsystem coupled to a display

and also employing transacted service, wherein the data structures have referential integrity and temporary copies of data structures are not created during editing of the data structures, the method comprising the steps of:

presenting on the display a representation of a plurality of data structures; and

providing a plurality of functions for either or both of naming and manipulation of data structures, the plurality of functions entirely excluding a save-as function so as to be incapable of being executed.

21. (Original) The method according to claim 20, wherein the plurality of manipulation functions comprises providing close, discard and rename functions if a newly-created data structure is being edited.

22. (Original) The method according to claim 20, wherein the plurality of manipulation functions comprises providing close and copy functions if an existing data structure is being edited.

23. (Original) The method according to claim 20, wherein the step of presenting on the display a representation of a plurality of data structures comprises presenting a graphical representation of a plurality of data structures.